Secrets
ABAP
Apex
C
C++
CloudFormation
COBOL
C#
CSS
Flex
Go
HTML
Java
JavaScript
Kotlin
Kubernetes
Objective C
PHP
PL/I
PL/SQL
Python
RPG
Ruby
Scala
Swift
Terraform
Text
TypeScript
T-SQL
VB.NET
VB6
XML

CSS static code analysis

Unique rules to find Bugs and Code Smells in your CSS code

All rules 25   🐞 Bug 16   ☢ Code Smell 9

Tags ⌄   Search by name...

Single line comment syntax should not be used

🐞 Bug

CSS properties should be valid

🐞 Bug

Units should be valid

🐞 Bug

"calc" operands should be correctly spaced

🐞 Bug

Color definitions should be valid

🐞 Bug

Expressions within "calc" should be valid

🐞 Bug

Selectors should be known

🐞 Bug

Shorthand properties that override related longhand properties should be avoided

🐞 Bug

"linear-gradient" directions should be valid

🐞 Bug

"at-rules" should be valid

🐞 Bug

Media features should be valid

🐞 Bug

Pseudo-element selectors should be valid

🐞 Bug

Pseudo-class selectors should be valid

🐞 Bug

"at-rules" should be valid   **Analyze your code**

🐞 Bug   🔺 Major ⍰

The W3C specifications define the valid `at-rules`. Only the official and browser-specific `at-rules` should be used to get the expected impact in the final rendering.

Noncompliant Code Example

```
@encoding "utf-8";
```

Compliant Solution

```
@charset "utf-8";
```

Available In:

sonarcloud ☁ | sonarqube

Properties should not be duplicated

🐞 Bug

"!important" should not be used on "keyframes"

🐞 Bug

Font declarations should contain at least one generic font family

🐞 Bug

CSS files should not be empty

☢ Code Smell

Selectors should not be duplicated

☢ Code Smell

Empty blocks should be removed

☢ Code Smell

Strings should not contain new lines

☢ Code Smell

Duplicated font names should be removed

☢ Code Smell

Multi-line comments should not be empty

☢ Code Smell

Duplicate imports should be removed

☢ Code Smell

Extra semicolons should be removed

☢ Code Smell

Selectors of lower specificity should